

# Guess a number puzzle

How many guesses do you need to **guarantee** to get a number between 1 and 100 if you're given "higher" or "lower" feedback?

1-20 is 5 guesses

How many for 1-100?

# Guess a number puzzle

{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,  
17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29,  
30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42,  
43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55,  
56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68,  
69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81,  
82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94,  
95, 96, 97, 98, 99, 100}

# Guess a number puzzle

Behind this is a really important computing **algorithm** that helps computers and the internet work quicker

It's called a **binary search algorithm**

# Guess a number puzzle

A **binary search** only works if the numbers are **in order**

Each time you guess you have to be able to say if the number is higher or lower. This lets you **remove half** of the remaining numbers each time

This means you **very quickly** get down to just a few numbers. So it's a really **efficient** way to search

# Guess a number puzzle

The other way to search is to start at the beginning and work your way through one number at a time

This is called a **linear search** - it works in a line

It's easier to do a linear search, but not as efficient if there are lots of numbers. You might have to do a linear search if the values **aren't in order**